Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression

Hamid Rezatofighi[1,2] Nathan Tsoi[1] JunYoung Gwak[1] Amir Sadeghian[1,3]
Ian Reid[2] Silvio Savarese[1]

[1]Computer Science Department, Stanford University, United states
[2]School of Computer Science, The University of Adelaide, Australia
[3]Aibee Inc, USA
hamidrt@stanford.edu

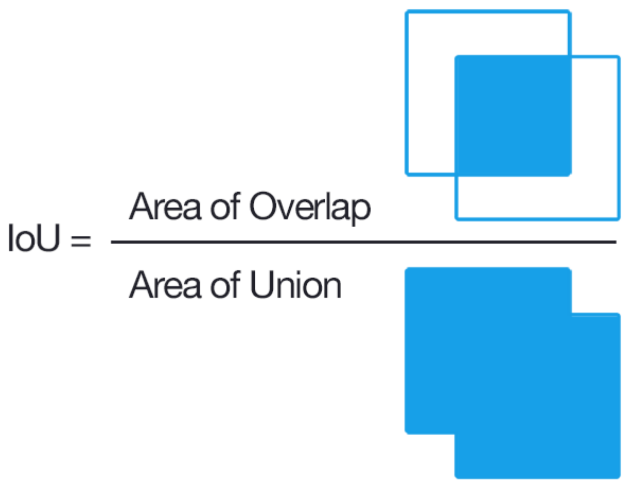
김성철

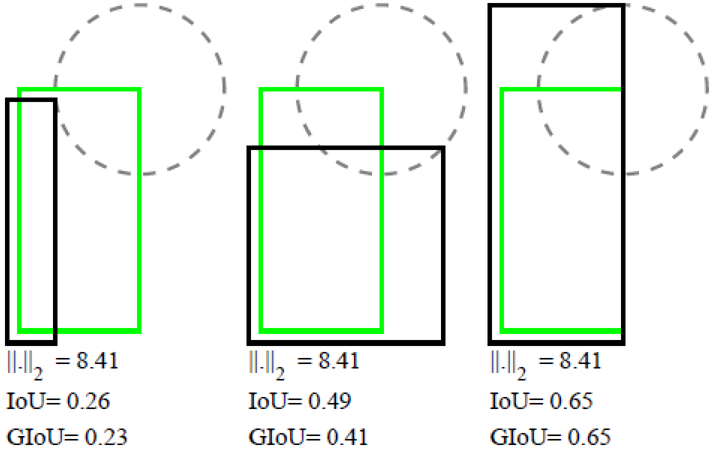
Contents

Introduction

- Loss function은 L_1, L_2-norms / Metric은 IoU 사용
  - **L_n-norms는 높여야 할 IoU와 strong correlation이 없음!**

  → IoU와 직접적인 관계가 있는 loss function을 제시





(a)

Introduction

- 두 물체가 겹치지 않는다면

→ IoU = 0

→ 두 물체가 얼마나 떨어져 있는지 모름!

→ <u>IoU의 약점을 보완하는 loss function 제시</u>

1. IoU와 같은 definition
2. scale invariant property 유지
3. 겹치는 물체간 strong correlation

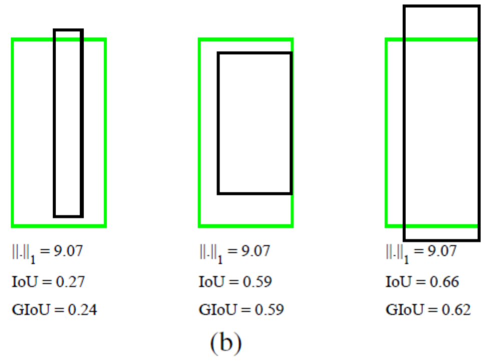


Figure 1. Two sets of examples (a) and (b) with the bounding boxes represented by (a) two corners (x_1, y_1, x_2, y_2) and (b) center and size (x_c, y_c, w, h). For all three cases in each set (a) ℓ_2-norm distance, $||.||_2$, and (b) ℓ_1-norm distance, $||.||_1$, between the representation of two rectangles are exactly same value, but their IoU and $GIoU$ values are very different.

Generalized Intersection over Union

- **Intersection over Union ($\boldsymbol{IoU}$)**
  - Loss ($\mathcal{L}_{IoU} = 1 - IoU$)
  - Scale invariant (independent from the scale)
  - **If $|\boldsymbol{A} \cap \boldsymbol{B}| = \boldsymbol{0}, \boldsymbol{IoU}(\boldsymbol{A}, \boldsymbol{B}) = \boldsymbol{0}$**

    → 두 물체 간 거리가 반영되지 않음!

Generalized Intersection over Union

- Generalized Intersection over Union ($GIoU$)

  For two arbitrary convex shapes (volumes) $A, B \subseteq \mathbb{S} \in \mathbb{R}^n$

  1. Find the smallest convex shapes $C \subseteq \mathbb{S} \in \mathbb{R}^n$ enclosing both A, B
  2. Calculate a ratio between the volume (area) occupied by C excluding A, B
  3. Divide by the total volume (area) occupied by C
  4. Subtract this ratio from the IoU value

Algorithm 1: Generalized Intersection over Union

input : Two arbitrary convex shapes: $A, B \subseteq \mathbb{S} \in \mathbb{R}^n$

output: $GIoU$

1 For A and B, find the smallest enclosing convex object C, where $C \subseteq \mathbb{S} \in \mathbb{R}^n$

2 $IoU = \frac{|A \cap B|}{|A \cup B|}$

3 $GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$

Generalized Intersection over Union

- **Generalized Intersection over Union ($GIoU$)**
  - Loss $\mathcal{L}_{GIoU} = 1 - GIoU$
  - Scale invariant
  - $\forall A, B \subseteq \mathbb{S},\ GIoU(A,B) \leq IoU(A,B)$ → $\lim_{A \to B} GIoU(A,B) = IoU(A,B)$
  - $\forall A, B \subseteq \mathbb{S},\ -1 \leq GIoU(A,B) \leq 1$

    → if $|A \cup B| = |A \cap B|,\ GIoU = IoU = 1$

    → $\lim_{\frac{|A \cup B|}{|C|} \to 0} GIoU(A,B) = -1$

Algorithm 1: Generalized Intersection over Union

input : Two arbitrary convex shapes: $A, B \subseteq \mathbb{S} \in \mathbb{R}^n$

output: $GIoU$

1 For A and B, find the smallest enclosing convex object C, where $C \subseteq \mathbb{S} \in \mathbb{R}^n$

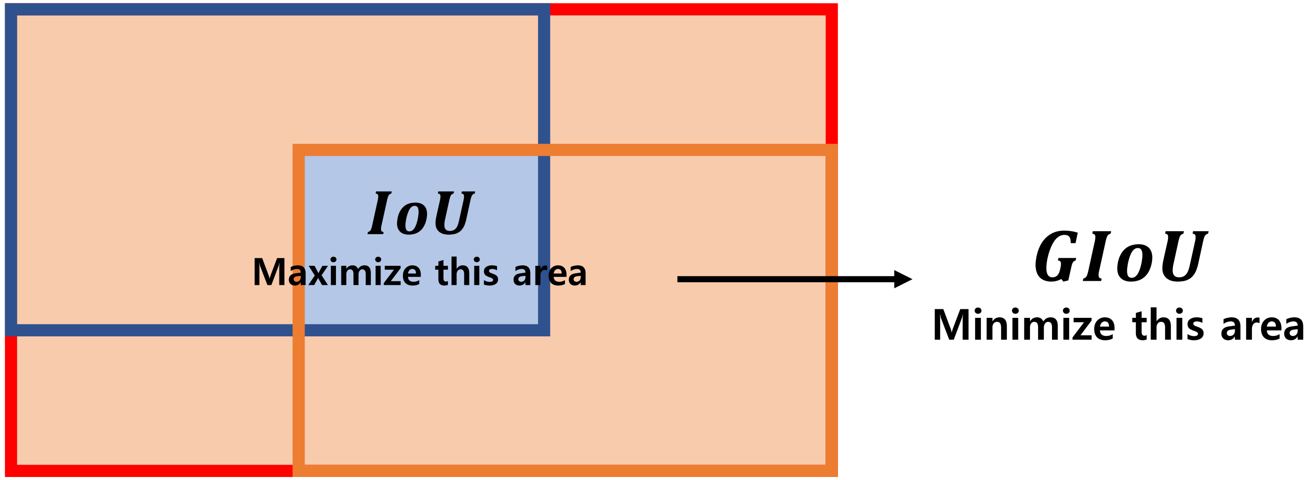
2 $IoU = \frac{|A \cap B|}{|A \cup B|}$

3 $GIoU = IoU - \frac{|C \backslash (A \cup B)|}{|C|}$

Generalized Intersection over Union

- ***GIoU*** **as Loss for Bounding Box Regression**
  - IoU or $GIoU$ can be directly used as a loss

    → 두 물체가 겹치지 않으면, *IoU* has zero gradient



Algorithm 2: IoU and $GIoU$ as bounding box losses

input : Predicted B^p and ground truth B^g bounding box coordinates:
$B^p = (x_1^p, y_1^p, x_2^p, y_2^p)$, $B^g = (x_1^g, y_1^g, x_2^g, y_2^g)$.

output: $\mathcal{L}_{IoU}$, $\mathcal{L}_{GIoU}$.

For the predicted box B^p, ensuring $x_2^p > x_1^p$ and $y_2^p > y_1^p$:
$\hat{x}_1^p = \min(x_1^p, x_2^p)$, $\hat{x}_2^p = \max(x_1^p, x_2^p)$,
$\hat{y}_1^p = \min(y_1^p, y_2^p)$, $\hat{y}_2^p = \max(y_1^p, y_2^p)$.

Calculating area of B^g: $A^g = (x_2^g - x_1^g) \times (y_2^g - y_1^g)$.

Calculating area of B^p: $A^p = (\hat{x}_2^p - \hat{x}_1^p) \times (\hat{y}_2^p - \hat{y}_1^p)$.

Calculating intersection $\mathcal{I}$ between B^p and B^g:
$x_1^{\mathcal{I}} = \max(\hat{x}_1^p, x_1^g)$, $x_2^{\mathcal{I}} = \min(\hat{x}_2^p, x_2^g)$,
$y_1^{\mathcal{I}} = \max(\hat{y}_1^p, y_1^g)$, $y_2^{\mathcal{I}} = \min(\hat{y}_2^p, y_2^g)$,

$$\mathcal{I} = \begin{cases} (x_2^{\mathcal{I}} - x_1^{\mathcal{I}}) \times (y_2^{\mathcal{I}} - y_1^{\mathcal{I}}) & \text{if} \quad x_2^{\mathcal{I}} > x_1^{\mathcal{I}}, y_2^{\mathcal{I}} > y_1^{\mathcal{I}} \\ 0 & \text{otherwise.} \end{cases}$$

Finding the coordinate of smallest enclosing box B^c:
$x_1^c = \min(\hat{x}_1^p, x_1^g)$, $x_2^c = \max(\hat{x}_2^p, x_2^g)$,
$y_1^c = \min(\hat{y}_1^p, y_1^g)$, $y_2^c = \max(\hat{y}_2^p, y_2^g)$.

Calculating area of B^c: $A^c = (x_2^c - x_1^c) \times (y_2^c - y_1^c)$.

$IoU = \frac{\mathcal{I}}{\mathcal{U}}$, where $\mathcal{U} = A^p + A^g - \mathcal{I}$.

$GIoU = IoU - \frac{A^c - \mathcal{U}}{A^c}$.

$\mathcal{L}_{IoU} = 1 - IoU$, $\mathcal{L}_{GIoU} = 1 - GIoU$.

Generalized Intersection over Union

- $GIoU$ as Loss for Bounding Box Regression
  - Strong correlation with IoU

    i.e. $IoU \leq 0.2$ and $GIoU \leq 0.2$

    → 끝에서는 $GIoU$가 더 가파른 gradient를 가짐

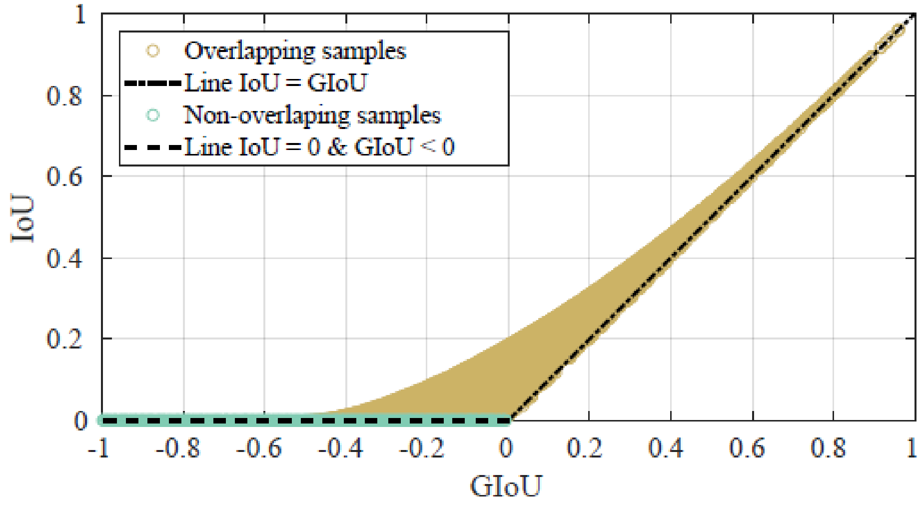


Figure 2. Correlation between GIoU and IOU for overlapping and non-overlapping samples.

Generalized Intersection over Union

- $\boldsymbol{GIoU}$ **as Loss for Bounding Box Regression**
  - Loss Stability

    Always $\boldsymbol{B^c} \geq \ \boldsymbol{B^g} \rightarrow \frac{\boldsymbol{A^c - \mathcal{U}}}{\boldsymbol{A^c}} > \mathbf{0}\ (\boldsymbol{A^c} \geq \boldsymbol{A^g}, \forall \boldsymbol{B^p} \in \mathbb{R}^{\mathbf{4}}$ **and** $\boldsymbol{A^g} \geq \mathbf{0})$

    $\boldsymbol{A^c} \geq \boldsymbol{\mathcal{U}}, \forall \boldsymbol{B^p} \in \mathbb{R}^{\mathbf{4}} \rightarrow \mathbf{0} \leq \boldsymbol{\mathcal{L}_{GIoU}} \leq \mathbf{2}, \forall \boldsymbol{B^p} \in \mathbb{R}^{\mathbf{4}}$

  - $\mathcal{L}_{GIoU}$ behavior when IoU = 0

    $\boldsymbol{\mathcal{L}_{GIoU}} = \mathbf{1} - \boldsymbol{GIoU} = \mathbf{1} + \frac{\boldsymbol{A^c - \mathcal{U}}}{\boldsymbol{A^c}} - \boldsymbol{IoU} = \mathbf{2} - \frac{\boldsymbol{\mathcal{U}}}{\boldsymbol{A^c}}$

    To minimize $\boldsymbol{\mathcal{L}_{GIoU}}$**, maximize** $\frac{\boldsymbol{\mathcal{U}}}{\boldsymbol{A^c}} \rightarrow \mathbf{0} \leq \frac{\boldsymbol{\mathcal{U}}}{\boldsymbol{A^c}} \leq \mathbf{1} \rightarrow$ **minimize** $\boldsymbol{A^c}$ **or maximize** $\boldsymbol{\mathcal{U}}\ (\boldsymbol{A^p})$

Experimental Results

- YOLO v3

Table 1. Comparison between the performance of **YOLO v3** [21] trained using its own loss (MSE) as well as $\mathcal{L}_{IoU}$ and $\mathcal{L}_{GIoU}$ losses. The results are reported on the **test set of PASCAL VOC 2007.**

Loss / Evaluation	AP		AP75	
	IoU	**GIoU**	**IoU**	**GIoU**
MSE [21]	.461	.451	.486	.467
$\mathcal{L}_{IoU}$	.466	.460	.504	.498
Relative improv %	1.08%	2.02%	3.70%	6.64%
$\mathcal{L}_{GIoU}$	**.477**	**.469**	**.513**	**.499**
Relative improv %	**3.45%**	**4.08%**	**5.56%**	**6.85%**

Table 2. Comparison between the performance of **YOLO v3** [21] trained using its own loss (MSE) as well as $\mathcal{L}_{IoU}$ and $\mathcal{L}_{GIoU}$ losses. The results are reported on 5K of the **2014 validation set of MS COCO**.

Loss / Evaluation	AP		AP75	
	IoU	**GIoU**	**IoU**	**GIoU**
MSE [21]	0.314	0.302	0.329	0.317
$\mathcal{L}_{IoU}$	0.322	0.313	0.345	0.335
Relative improv %	2.55%	3.64%	4.86%	5.68%
$\mathcal{L}_{GIoU}$	**0.335**	**0.325**	**0.359**	**0.348**
Relative improv %	**6.69%**	**7.62%**	**9.12%**	**9.78%**

Table 3. Comparison between the performance of **YOLO v3** [21] trained using its own loss (MSE) as well as using $\mathcal{L}_{IoU}$ and $\mathcal{L}_{GIoU}$ losses. The results are reported on the **test set of MS COCO 2018**.

Loss / Evaluation	AP	AP75
MSE [21]	.314	.333
$\mathcal{L}_{IoU}$	.321	.348
Relative improv %	2.18%	4.31%
$\mathcal{L}_{GIoU}$	**.333**	**.362**
Relative improv %	**5.71%**	**8.01%**

Experimental Results

- Faster R-CNN

Table 4. Comparison between the performance of **Faster R-CNN** [22] trained using its own loss (ℓ_1-smooth) as well as $\mathcal{L}_{IoU}$ and $\mathcal{L}_{GIoU}$ losses. The results are reported on the **test set of PASCAL VOC 2007**.

Loss / Evaluation	AP		AP75	
	IoU	**GIoU**	**IoU**	**GIoU**
ℓ_1-smooth [22]	.370	.361	.358	.346
$\mathcal{L}_{IoU}$	.384	.375	.395	.382
Relative improv. %	3.78%	3.88%	10.34%	10.40%
$\mathcal{L}_{GIoU}$	**.392**	**.382**	**.404**	**.395**
Relative improv. %	**5.95%**	**5.82%**	**12.85%**	**14.16%**

Table 5. Comparison between the performance of **Faster R-CNN** [22] trained using its own loss (ℓ_1-smooth) as well as $\mathcal{L}_{IoU}$ and $\mathcal{L}_{GIoU}$ losses. The results are reported on the **validation set of MS COCO 2018.**

Loss / Evaluation	AP		AP75	
	IoU	**GIoU**	**IoU**	**GIoU**
ℓ_1-smooth [22]	.360	.351	.390	.379
$\mathcal{L}_{IoU}$	.368	.358	.396	.385
Relative improv.%	2.22%	1.99%	1.54%	1.58%
$\mathcal{L}_{GIoU}$	**.369**	**.360**	**.398**	**.388**
Relative improv. %	**2.50%**	**2.56%**	**2.05%**	**2.37%**

Table 6. Comparison between the performance of **Faster R-CNN** [22] trained using its own loss (ℓ_1-smooth) as well as $\mathcal{L}_{IoU}$ and $\mathcal{L}_{GIoU}$ losses. The results are reported on the **test set of MS COCO 2018**.

Loss / Metric	AP	AP75
ℓ_1-smooth [22]	.364	.392
$\mathcal{L}_{IoU}$	**.373**	.403
Relative improv.%	**2.47%**	2.81%
$\mathcal{L}_{GIoU}$	**.373**	**.404**
Relative improv.%	**2.47%**	**3.06%**

Experimental Results

- Mask R-CNN

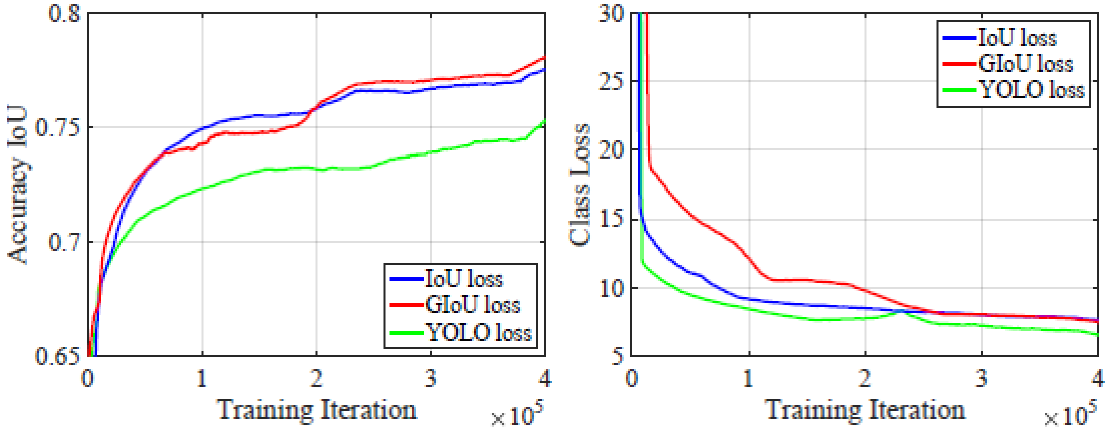
Table 7. Comparison between the performance of **Mask R-CNN** [6] trained using its own loss (ℓ_1-smooth) as well as $\mathcal{L}_{IoU}$ and $\mathcal{L}_{GIoU}$ losses. The results are reported on the **validation set of MS COCO 2018**.

Loss / Evaluation	AP		AP75	
	IoU	**GIoU**	**IoU**	**GIoU**
ℓ_1-smooth [6]	.366	.356	.397	.385
$\mathcal{L}_{IoU}$	.374	.364	.404	.393
Relative improv.%	2.19%	2.25%	1.76%	2.08%
$\mathcal{L}_{GIoU}$	**.376**	**.366**	**.405**	**.395**
Relative improv. %	**2.73%**	**2.81%**	**2.02%**	**2.60%**

Table 8. Comparison between the performance of **Mask R-CNN** [6] trained using its own loss (ℓ_1-smooth) as well as $\mathcal{L}_{IoU}$ and $\mathcal{L}_{GIoU}$ losses. The results are reported on the **test set of MS COCO 2018**.

Loss / Metric	AP	AP75
ℓ_1-smooth [6]	.368	.399
$\mathcal{L}_{IoU}$	**.377**	.408
Relative improv.%	**2.45%**	2.26%
$\mathcal{L}_{GIoU}$	**.377**	**.409**
Relative improv.%	**2.45%**	**2.51%**

Experimental Results



(a) (b)

Figure 3. The classification loss and accuracy (average IoU) against training iterations when YOLO v3 [21] was trained using its standard (MSE) loss as well as $\mathcal{L}_{IoU}$ and $\mathcal{L}_{GIoU}$ losses.

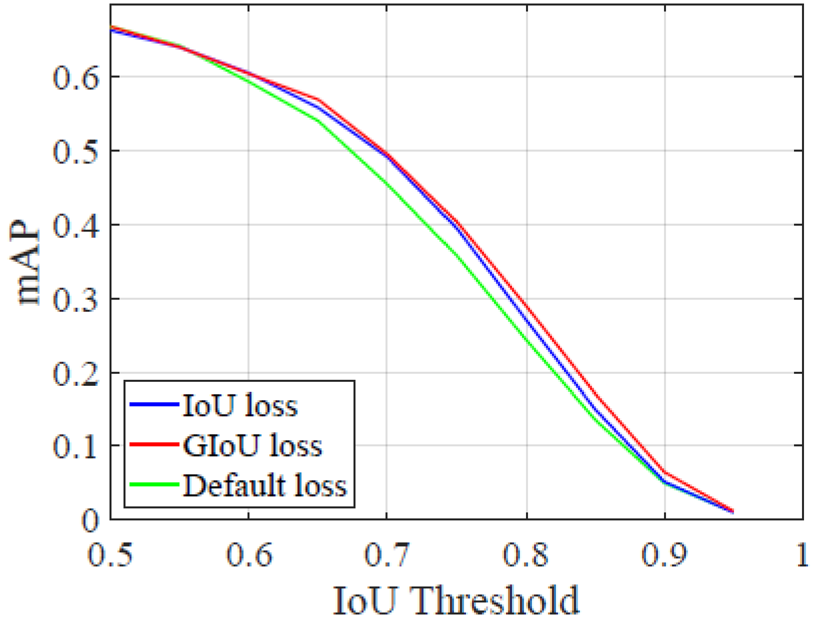


Figure 4. mAP value against different IoU thresholds, *i.e.* $.5 \leq IoU \leq .95$, for **Faster R-CNN** trained using ℓ_1-smooth (green), $\mathcal{L}_{IoU}$ (blue) and $\mathcal{L}_{GIoU}$ (red) losses.

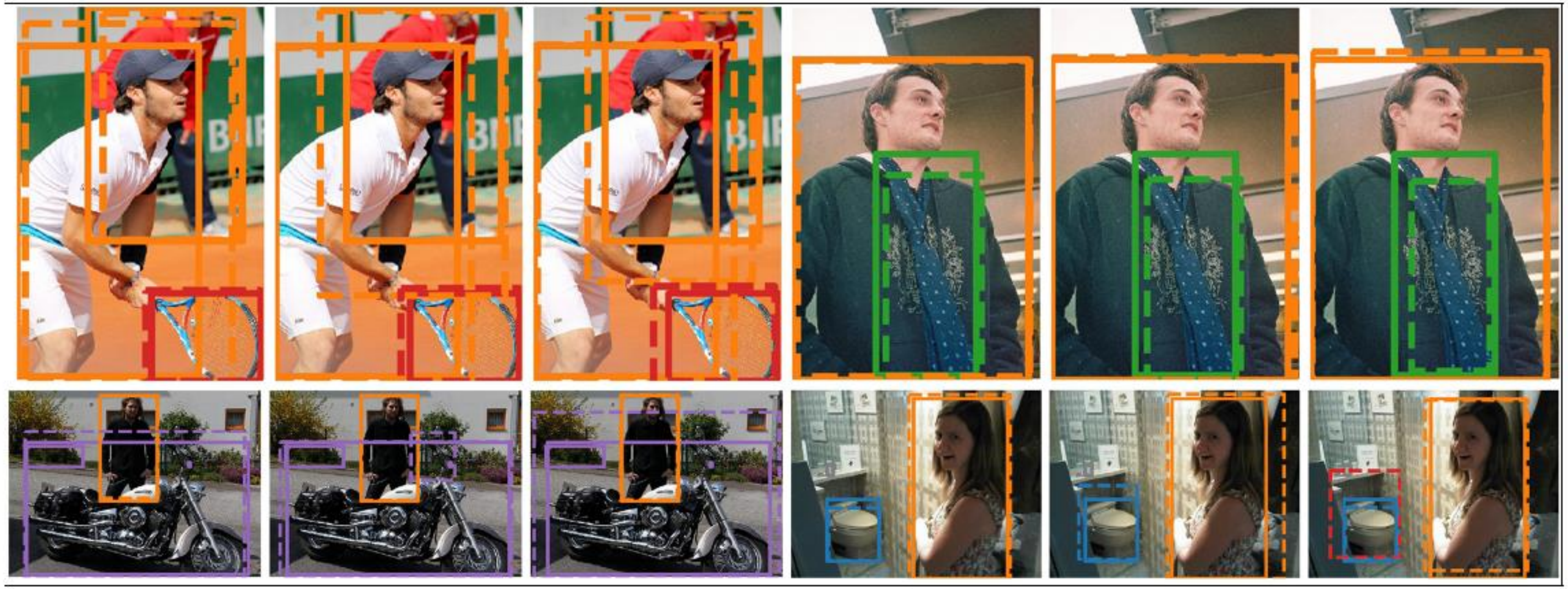


Figure 5. Example results from COCO validation using YOLO v3 [21] trained using (left to right) $\mathcal{L}_{GIoU}$, $\mathcal{L}_{IoU}$, and MSE losses. Ground truth is shown by a solid line and predictions are represented with dashed lines.

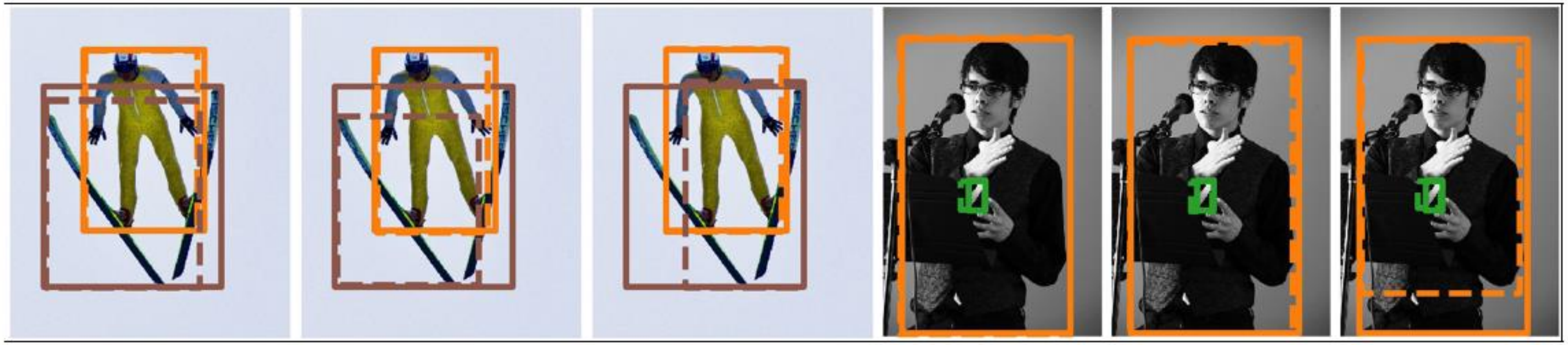


Figure 6. Two example results from COCO validation using Mask R-CNN [6] trained using (left to right) $\mathcal{L}_{GIoU}$, $\mathcal{L}_{IoU}$, ℓ_1-smooth losses. Ground truth is shown by a solid line and predictions are represented with dashed lines.

Conclusions

- IoU의 약점을 보안한 $GIoU$ 제안
- $GIoU$를 bounding box regression loss로써 사용
- 회전 도형과 3D object detection에도 적용 예정

https://github.com/generalized-iou

감 사 합 니 다